

ARGUMENT

Claims 1-34 are pending in the application. All the independent claims have been amended, more specifically, claims 1, 9-14, 21, and 27-34 are amended to better define the claims. Claim 8 is canceled.

§101 Rejection

Claims 27-34 were rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter. Specifically, the claims were objected to as possibly covering a carrier wave. Accordingly, claims 27-33 have been canceled and claim 34 has been amended to call for a storage medium which would exclude a carrier wave. Accordingly, the applicant believes the section 101 rejection has been addressed and should be withdrawn.

§102(e) Rejection

Claims 1 -34 are rejected under 35 U.S.C. §102(e) as being anticipated by Shiomi et al., U.S. Publication number 2006 0095919 A1 (“Shiomi”). Shiomi describes a Java middleware unit 33 which includes a virtual machine (VM, element 33a in Fig. 15), an application managing unit (element 33b in Fig. 15) and a class library storing unit (element 33c in Fig. 15). It also has an OS unit 34 that has a kernel (element 34a) and a hardware unit that has a CPU (element 35a).

In order to anticipate as claim, all the elements of the claim must be found in a single reference. All the independent claims have been amended to further clarify the claims.

CLAIM 1

In all of the pending independent claims, a determination is made whether a process is a kernel process. The process is then matched to a first or second predefined process. The kernel usage of the process is then reviewed and if the kernel usage is above a threshold, the process is aborted.

In Shiomi, the Virtual Machine controls the execution of the application and the kernel. The kernel is described in Shiomi as follows:

The kernel 34a manages resources provided to applications, in units of tasks corresponding to the applications. Specifically, the kernel 34a generates a task corresponding to an application and the first thread of the task in response to an instruction from the application managing unit 33b, and reserves resources necessary for the operation of the VM unit 33a using the first thread. The kernel 34a also generates threads required by the application, provides resources required by the application using the generated threads, and holds resource names of the

provided resources and a task ID of a task to which the threads belong, in correspondence. When notified by the application managing unit 33a of a task ID corresponding to an application which is completed, the kernel 34a collects all resources specified by resource names held in correspondence with the notified task ID.

Shiomi fails to disclose determining if a process is a kernel process, fails to disclose that the process is matched to a first or second process and third, the kernel usage is not compared to a threshold and forth, if the kernel is over the threshold, the process is terminated. Not only does the claimed method track kernel usage, it aborts processes that have excessive usage. The result is a better operating system that is an advancement over the prior art. Accordingly, as these claimed elements are not present in Shoimi or in any other cited art, the claims should be allowed. As claims 2- 7 are dependent on claim 1 and an element is missing from claim 1, this element is missing from claims 2-7 and claims 2-7 also are not anticipated.

CLAIM 14

Similar to claim 1, Shiomi fails to disclose determining if a process is a kernel process, fails to disclose that the process is matched to a first or second process and third, the kernel usage is not compared to a threshold and forth, if the kernel is over the threshold, the process is terminated. Not only does the claimed method track kernel usage, it aborts processes that have excessive usage. The result is a better operating system that is an advancement over the prior art. Accordingly, as these claimed elements are not present in Shoimi or in any other cited art, the claims should be allowed. As claims 15-20 are dependent on claim 14 and an element is missing from claim 14, this element is missing from claims 15-20 and claims 15-20 also are not anticipated.

CLAIM 21

Shiomi fails to disclose determining if a process is a kernel process, fails to disclose that the process is matched to a first or second process and third, the kernel usage is not compared to a threshold and forth, if the kernel is over the threshold, the process is terminated. Not only does the claimed method track kernel usage, it aborts processes that have excessive usage. The result is a better operating system that is an advancement over the prior art. Accordingly, as these claimed elements are not present in Shoimi or in any other cited art, the claims should be allowed. As claims 22-24 are dependent on claim 21 and an

element is missing from claim 21, this element is missing from claims 22-24 and claims 22-24 also are not anticipated.

CLAIM 25

Claim 25 calls for a flag to indicate whether the process is a first defined process or a second defined process. Again, Shiomi does not make such a determination or use such a flag. The Office action sites to the same sections of Shiomi for disclosing this element. The Applicant refers to the previous paragraphs for a discussion of the cited sections and the failure to disclose the element of making the determination of the type of process. As claim 26 is dependent on claim 25 and an element is missing from claim 25, this element is missing from claim 26 and claim 26 also is not anticipated.

CLAIM 27

Claim 27 calls for a data structure that is a hash table with a first field identifying user processes and a second field identifying kernel resources allocated to the user processes. A hash table is briefly mention in Shiomi at paragraph 142:

When a listener is registered, a class library unit calls the application managing unit 33b and acquires an application ID of an application which registers the listener. The application managing unit 33b specifies the application ID and provides it to the class library unit, in the following manner. The application managing unit 33b acquires an instance of a thread class which has called the application managing unit 33b, and specifies an ID (thread ID) of the instance which has called the application managing unit 33b, through the use of a function "hashcode" that is a method for retrieving a hashcode value from the instance. The application managing unit 33b then specifies an application ID corresponding to the thread ID, with reference to a table that shows the correspondence between application IDs and thread IDs. The application managing unit 33b provides the specified application ID to the class library unit. The hashcode value retrieved here is an ID assigned to each instance so that the VM unit 33a can efficiently manage instances generated from classes. The class library unit stores a combination of the application ID provided from the application managing unit 33b, the thread ID of the generated special thread, and the listener ID of the registered listener, in a table.

Shiomi does not disclose **creating** a hash table as described in the claim (Shiomi describes using a pre-existing hash table) and this element is missing. The use of a hash table would allow the usage data to be shared in a smaller and more secure format which is an

advancement over the prior art. As claims 28-29 are dependent on claim 27 and an element is missing from claim 27, this element is missing from claims 28-29 and claims 28-29 also are not anticipated.

CLAIM 30

Claim 30 calls for determining whether the process is a kernel process or a user process and storing the process determination tag as an entry in a hash table. Shiomi does not disclose **creating** a hash table as described in the claim (Shiomi describes using a pre-existing hash table) and this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated. In addition, the use of a hash table would allow the usage data to be shared in a smaller and more secure format which is an advancement over the prior art.

CLAIM 31

Claim 31 calls for determining whether the process is a kernel process or a user process and storing the process determination tag as an entry in a hash table. Shiomi does not disclose **creating** a hash table as described in the claim (Shiomi describes using a pre-existing hash table) and this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated. In addition, the use of a hash table would allow the usage data to be shared in a smaller and more secure format which is an advancement over the prior art.

CLAIM 32

Claim 32 calls for determining whether the process is a kernel process or a user process and storing the process determination tag as an entry in a hash table. Shiomi does not disclose **creating** a hash table as described in the claim (Shiomi describes using a pre-existing hash table) and this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated. In addition, the use of a hash table would allow the usage data to be shared in a smaller and more secure format which is an advancement over the prior art.

CLAIM 33

Claim 33 calls for identifying an amount of kernel resources allocated to a process indicated by a selected tag and storing the amount as an entry in a hash table. Shiomi does not disclose **creating** a hash table as described in the claim (Shiomi describes using a pre-existing hash table) and this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated. In addition, the use of a hash table would allow the usage data to be shared in a smaller and more secure format which is an advancement over the prior art.

CLAIM 34

Claim 34 calls for a flag indicating whether the process is a first defined process or a second defined process. Similar to claim 1, as Shiomi does not differentiate between a first process and a second process as called for in the claims, this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated.

REMARKS

In view of the above amendment, applicant believes the pending application is in condition for allowance. If the Examiner has any questions or suggestions, the Examiner is encouraged to call the applicant direct at 312-474-6610.

Dated: September 21, 2007

Respectfully submitted,

By___/W. J. Kramer/_____
William J. Kramer

Registration No.: 46,229
MARSHALL, GERSTEIN & BORUN
233 S. Wacker Drive, Suite 6300
Sears Tower
Chicago, Illinois 60606-6357
(312) 474-6300
Attorney for Applicant